AFRL-RY-WP-TR-2016-0123

MATHEMATICS OF SENSING, EXPLOITATION, AND EXECUTION (MSEE)

Hierarchical Representations for the Evaluation of Sensed Data

Stuart Geman
Brown University

JUNE 2016
Final Report

STINFO COPY

AIR FORCE RESEARCH LABORATORY
SENSORS DIRECTORATE
WRIGHT-PATTERSON AIR FORCE BASE, OH 45433-7320
AIR FORCE MATERIEL COMMAND
UNITED STATES AIR FORCE

NOTICE AND SIGNATURE PAGE

Using Government drawings, specifications, or other data included in this document for any purpose other than Government procurement does not in any way obligate the U.S. Government. The fact that the Government formulated or supplied the drawings, specifications, or other data does not license the holder or any other person or corporation; or convey any rights or permission to manufacture, use, or sell any patented invention that may relate to them.

This report is the result of contracted fundamental research deemed exempt from public affairs security and policy review in accordance with SAF/AQR memorandum dated 10 Dec 08 and AFRL/CA policy clarification memorandum dated 16 Jan 09. This report is available to the general public, including foreign nationals.

AFRL-RY-WP-TR-2016-0123 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

// Signature//

Jared Culbertson, Program Manager
Electro-Optic Exploitation Branch
Layered Sensing Exploitation Division

// Signature//

Clare Mikula, Branch Chief
Electro-Optic Exploitation Branch
Layered Sensing Exploitation Division

// Signature//

Doug Hager, Deputy
Layered Sensing Exploitation Division
Sensors Directorate

This report is published in the interest of scientific and technical information exchange, and its publication does not constitute the Government's approval or disapproval of its ideas or findings.

*Disseminated copies will show "//Signature//" stamped or typed above the signature blocks.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
June 2016	Final	14 September 2011 – 30 December 2015

4. TITLE AND SUBTITLE
MATHEMATICS OF SENSING, EXPLOITATION, AND EXECUTION (MSEE)
Hierarchical Representations for the Evaluation of Sensed Data

5a. CONTRACT NUMBER
FA8650-11-1-7151

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
61101E

6. AUTHOR(S)
Stuart Geman

5d. PROJECT NUMBER
1000

5e. TASK NUMBER
N/A

5f. WORK UNIT NUMBER
Y0PP

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Brown University
1 Prospect Street
Providence, RI 02912-9079

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Air Force Research Laboratory
Sensors Directorate
Wright-Patterson Air Force Base, OH 45433-7320
Air Force Materiel Command
United States Air Force

Defense Advanced Research Projects Agency/DARPA/DSO
675 N Randolph Street
Arlington, VA 22203

10. SPONSORING/MONITORING AGENCY ACRONYM(S)
AFRL/RYAT

11. SPONSORING/MONITORING AGENCY REPORT NUMBER(S)
AFRL-RY-WP-TR-2016-0123

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

13. SUPPLEMENTARY NOTES
This report is the result of contracted fundamental research deemed exempt from public affairs security and policy review in accordance with SAF/AQR memorandum dated 10 Dec 08 and AFRL/CA policy clarification memorandum dated 16 Jan 09. This report is available to general public including foreign nationals. This material is based on research sponsored by Air Force Research laboratory (AFRL) and the Defense Advanced Research Agency (DARPA) under agreement number FA8650-11-1-7151. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation herein. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies of endorsements, either expressed or implied, of Air Force Research Laboratory (AFRL) and the Defense Advanced Research Agency (DARPA) or the U.S. Government. Report contains color.

14. ABSTRACT
The primary goal of this project was to build fully generative hierarchical scene models and accompanying algorithms and software for inference from still imagery. A secondary goal was to develop a feasible approach to learning these scene models from data. Other goals were less central, but included making connections and contributing to theories of the mammalian visual system, and exploiting descriptive text that may accompany a still image for improved inference. The focus of the Brown team was on single images of street scenes; there was no intention to work with frame sequences. The MSEE goals were ambitious, as were ours. Certainly we failed to meet them and, in fact, our four-or-so year effort can be described as an expedition with a continuously narrowing objective. At the same time, we would suggest that a critical piece of a structure that can support scalable human-level performance has been put in place, new and useful computational tools were discovered, and a new approach to testing vision systems, that places relationships and attributes at the same level of importance as identification, was developed.

15. SUBJECT TERMS
scene understanding, bayesian networks, context-sensitive grammar, restricted turing test, computer vision, semantic description, street scenes, belief propagation, generative models, nonlinear filtering, sufficient statistics

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT:	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON (Monitor)
a. REPORT	b. ABSTRACT	c. THIS PAGE			Jared Culbertson
Unclassified	Unclassified	Unclassified	SAR	20	19b. TELEPHONE NUMBER *(Include Area Code)* N/A

Hierarchical Representations for the Evaluation of Sensed Data

Final Report

Mathematics of Sensing, Exploitation and Execution

Defense Advanced Research Projects Agency

03/06/2016

1 Summary

The primary goal of this project was to build fully generative hierarchical scene models and accompanying algorithms and software for inference from still imagery. A secondary goal was to develop a feasible approach to learning these scene models from data. Other goals were less central, but included making connections and contributing to theories of the mammalian visual system, and exploiting descriptive text that may accompany a still image for improved inference. The focus of the Brown team was on single images of street scenes; there was no intention to work with frame sequences.

An unanticipated project emerged after a great deal of discussion at and following the kickoff meeting about methods for evaluating vision systems. There was general agreement that existing ROC-based performance metrics were not well matched to the goals of the MSEE program, which were more about scene understanding than object detection. Following several months of discussion, the Brown team proposed an outline of a "Restricted Turing Test," and was asked to devote resources to an investigation of feasibility.

Concerning the Turing test, the team was lead to a substantive problem and a solution which we believe has the potential to "raise the bar" in computer vision and encourage the development of systems displaying deeper, semantic-level, analyses of images. A prototype system was built, which lead to a Ph.D. thesis, a paper in the Proceedings of the National Academy of Sciences, and ongoing work on scene models and the evaluation of scene-analysis systems. Although our efforts focused on evaluating systems designed to parse street scenes (see §2), the same evaluation approach extends to video.

Concerning the main objective of building a fully generative model and an associated inference engine, we began the project with the assumption that the most challenging and fundamental task would be in defining a coherent, "context-sensitive," grammar–that is, a recursive set of *composition* rules that could, in general, depend upon the detailed, and unforeseeable, *content* of the constituents being composed. (An extreme example is the recognition of two entities as "the same."). We ended the project with a very different focus, having encountered what we believe to be an unavoidable and very challenging technical barrier to building scalable generative vision systems.

In brief, the challenge is to find a coherent model for a decidedly non-grammar-like feature of biological vision: the over-representation of latent variables and pixel-level inputs, due to the multiple roles played by a given entity in anything like a human semantic-level description of a scene. To appreciate the problem, consider for instance that a leaf can be seen as a taxonomy-specific shape with taxonomy-specific stem structure, and a season-dependent color, by simultaneous reference to the same regions of the image. A face can be seen coarsely and finely, and characterized as smooth, tired, part of a continuation of the neck, and weather-worn, all at the same time. Spatial segmentation based on these semantic characteristics is plainly impossible. The same applies to abstractions—i.e. latent representations of parts and objects. These can participate in the representations of multiple compositions, simultaneously. A hand may be part of a pair of hands belonging to two individuals under the composition holding hands, and a continuation of a wrist via the composition of a

forearm, simultaneously. Here too, segmentation, whereby each variable is given a unique assignment as part of a particular composition, is unnatural, at best, and most likely a barrier to scalable accurate performance.

In short, tree-like structures, in which each part or pixel belongs uniquely to a single composition are inadequate. This is not a problem in discriminant models, e.g. convolution nets produce massively overlapping representations. But traditional approaches to generative part-based models either involve an artificial segment or include an ad-hoc compensation for a"double counting."

This comes off as esoteric. But consider the following thought experiment: suppose we had a model for the distribution of image patches that contain a pedestrian, as defined by a very large (say infinite) ensemble of street scenes. More carefully, imagine having a *likelihood ratio* for any given image patch–the ratio of the likelihood of the pixel data under the hypothesis that it contains a pedestrian to the likelihood under the single alternative that the patch does not contain a pedestrian. Suppose further that we could feasibly evaluate this ratio for *every* patch of every size in the image. And finally, suppose that thresholding the likelihood ratio gave superior (near human) ROC performance. Then the task, recognizing pedestrians, would become a purely computational problem, albeit a very challenging one.

We note that

1. Although we can not be certain, we have evidence that we can build such a model and, furthermore, from a surprisingly small amount of data (hundreds or a few thousand examples);
2. The model is extensible, meaning that if it is too weak we can add new features as needed–e.g. models of heads or hands or limbs, models of hair, or lack of hair–without rebuilding any part of the existing model;
3. The essential feature of the approach is the ability to avoid any kind of explicit segmentation; features and parts can overlap without loosing normalization of the likelihood ratios. In short, more is better.

Some of the details are in §3 & §4.

There is nothing, *per se*, that limits these models from being hierarchical, as in labeling two pedestrians as "walking together" and/or "holding hands," and so-on.

Of course the computational problem is by no means a side issue. Indeed, it has been argued that the *existence* of GPU's, developed largely for gaming, has done more to shape current state-of-the-art computer vision algorithms, namely convolution neural networks, than any biologically faithful model of the visual cortices. One approach to computing in a probabilistic model is through "loopy" belief propagation. The Brown team has recently discovered some variations that work well with hierarchical, parts-based models, as described in §3. The results clearly demonstrate the power of context, as captured by a hierarchy of *relational* compositions.

The MSEE goals were ambitious, as were ours. Certainly we failed to meet them and, in fact, our four-or-so year effort can be described as an expedition with a continuously narrowing objective. At the same time, we would suggest that a critical piece of a structure that can support scalable human-level performance has been put in place, new and useful computational tools were discovered, and a new approach to testing vision systems, that places *relationships* and *attributes* at the same level of importance as identification, was developed.

2 Restricted Turing Test

Consider the task of building a semantic description of Figure (1). Note that the two closest people are walking together, and that the older pair, in front of them, are standing and talking. Note also the two rows of red chairs, most of which are largely occluded. Nevertheless, we know a great deal about the shape and colors of the chairs

Figure 1: Urban street scene

that are furthest from the camera, though almost entirely blocked. In all of these cases, and routinely in real images, it is the *context*, represented as a *composition*, that allows us to make conclusions about the parts.

Compositions are about relationships among parts. How do we test for, and thereby encourage, relational reasoning? The usual approach to vision challenges won't work here: preparing and scoring results with fully relationally labeled test sets is infeasible. We propose, instead, a "restricted Turing test."

We start with an application-specific vocabulary of objects, attributes, and relationships, and we construct a "query engine" that automatically serves up binary questions about any selected image. The test, then, is a sequence of such queries, designed to probe the richness of a vision system's representation of the scene. The preparation of the test involves a human, who declares the answer to the proposed question as either true, false, or ambiguous. Ambiguous questions do not make it to the list. The test construction process continues, iteratively, until the list of "suitable" questions is exhausted, at which point the engine quits.

A selection of questions from a test prepared by a prototype query engine is shown in Figure (2). Answers, including identifying Q24 as ambiguous, are provided by the operator ("Human in the Loop"). Localizing questions include, implicitly, the qualifier "partially visible in the designated region" and instantiation (existence and uniqueness) questions implicitly include "not previously instantiated." The localizing windows used for each of the four instantiations (vehicle 1, person 1, person 2, and person 3) are indicated by the colored rectangles (blue, thick border; red, thin border; and yellow, dashed border). The colors are included in the questions for illustration. In the actual test, each question designates a single rectangle through its coordinates, so that "Is there a unique person in the blue region?" would actually read "Is there a unique person in the designated region?".

The actual administration of the test is fully automatic. Questions are posed in a simple syntax, the system under test delivers a "yes" or "no" answer, the system is given the correct answer, and the next question is posed.

The key of course is the query engine. The overarching "design principle" is unpredictability. As already noted, the test is constructed iteratively: Before producing question $k+1$, let $H = [(q_1, q_2), \ldots, (q_k, x_k)]$, where $q_i \in \mathcal{Q}$ is any syntactically allowed question (restricted to the aforementioned vocabulary), and where

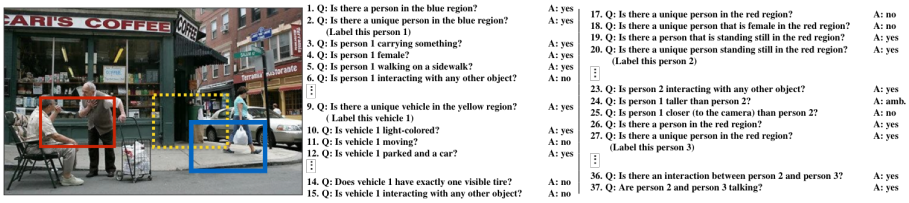

Figure 2: Sample questions from a restricted Turing test

$x_i \in \{0, 1\}$ is the human-provided *true* answer to q_i. In other words, let H is the history of questions and correct answers up to this point in the test preparation. The engine, then, is a function that takes any H and produces a new history with one additional query: $H \rightarrow [(q_1, q_2), \ldots, (q_k, x_k), (q, x)]$. The engine is trained to produce approximately unpredictable questions:

$$P_H(X_q = x) = \frac{P\{I : H(I) = 1, X_q(I) = x\}}{P\{I : H(I) = 1\}} \approx 0.5$$

Here, I represents a random image from the ensemble of interest (urban street scenes in the prototype system) and $H(I) = 1$ means that the image I satisfies the history. The probabilities can be estimated from a training set of sample images, using simple empirical frequencies (as in the prototype system), or parametrically using scene models (as in the ongoing research effort). The important point is that the answer is essentially unpredictable from the sequence of correct answers to the already-delivered questions. There is no "gaming" the system–i.e. the only relevant information to the system under test is the image itself.

Two further design characteristics of the query engine are worth highlighting. One is that the loop structure of the algorithm is constructed so as to prefer "story lines"–subsequences of questions about already instantiated objects. Refer again to the example, once Person 1 and Person 2 are instantiated, their possible relationships are explored, and exhausted, after which new instantiation questions establish the uniquely identified "Person 3". An ensuing sequence finally establishes that Person 2 and Person 3 are talking. Even with the limited vocabulary and restricted syntax used in the prototype, there are an enormous number of available queries. Story lines serve to promote questions about relationships and attributes, which goes well beyond detection, *per se*.

Additionally, there is a random element in the choice of questions, such that a question is chosen at random from the collection of questions that are (i) essentially equally unpredictable and (ii) can be found at the same depth within a story line. Multiple runs produce multiple tests on the same image.

Finally, we note that test preparation does not require exhaustive off-line labeling. Once the engine is trained, the human role is, in essence, to take the test ("just-in-time truthing"), which of course is nearly effortless.

Many design choices were made, some more or less arbitrarily, and much needs to be done in order to scale to larger vocabularies and more general scenarios. These and other issues are discussed in detail in [14] and the accompanying supplement, and the project continues [26,27].

3 Belief Propagation in Stochastic Scene Grammars

How important is context? One way to get at this is to build a grammar-like system (in this case, a Bayes Net), representing a hierarchy of part-whole relationships, and a simple data model based on HOG filters for continuous-valued pixels, or independent Gaussian variables for (noisy) line drawings. We outline here

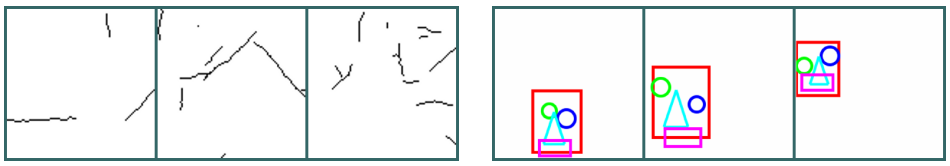

Figure 3: Left: Samples of contour maps generated by our grammar model of curves. Right: Samples of scenes with a single face generated by our grammar model of faces.

two examples, one identifying boundaries and the other for face detection. Each system is a realization of a probabilistic grammar on what we call "bricks," which are abstract latent variables. The computational engine is a variation on loopy belief propagation. Most of the details can be found in [28] (which is in review and not yet available for distribution).

In the models we consider, every object has a type from a finite alphabet and a pose from a finite but large pose space. While classical language models generate sentences using a single derivation, the grammars we consider generate scenes using multiple derivations. These derivations can be unrelated or they can share sub-derivations. This allows for very general descriptions of scenes.

We show how to represent the distributions defined by probabilistic scene grammars using factor graphs, and this opens the door to loopy belief propagation (LBP) for approximate inference. Inference with LBP simultaneously combines "bottom-up" and "top-down" contextual information. For example, when faces are defined using a composition of eyes, nose and mouth, the evidence for a face or one of its parts provides contextual influence for the whole composition. Inference via message passing naturally captures chains of contextual evidence. LBP also naturally combines multiple contextual cues. For example, the presence of an eye may provide contextual evidence for a face at two different locations because a face has a left and a right eye. However, the presence of two eyes side by side provides strong evidence for a single face between them.

We demonstrate the practical feasibility of the approach on two very different applications: curve detection and face localization. Figure 3 shows samples from the two different grammars we use for the experimental results. The contributions here include (1) a unified framework for contextual modeling that can be used in a variety of applications; (2) a construction that maps a probabilistic scene grammar to a factor graph together with an efficient message passing scheme; and (3) experimental results showing the effectiveness of the approach.

Model. Scenes are defined using a library of building blocks, or *bricks*, that have a type and a pose. Bricks are generated spontaneously or through expansions of other bricks. This leads to a hierarchical organization of the elements of a scene.

Definition 3.1. *A* probabilistic scene grammar $\mathcal{G}$ *consists of*

1. *A finite set of symbols, or types,* Σ.
2. *A finite pose space,* Ω_A*, for each symbol* $A \in \Sigma$.
3. *A finite set of production rules,* $\mathcal{R}$*. Each rule* $r \in \mathcal{R}$ *is of the form* $A_0 \to \{A_1, \ldots, A_{N_r}\}$*, where* $A_i \in \Sigma$*. We use* $\mathcal{R}_A$ *to denote the rules with symbol* A *in the left-hand-side (LHS). We use* $A_{r,i}$ *to denote the* i*-th symbol in the right-hand-side (RHS) of a rule* r.
4. *Rule selection probabilities,* $P(r)$*, with* $\sum_{r \in \mathcal{R}_A} P(r) = 1$ *for each symbol* $A \in \Sigma$.
5. *For each rule* $r = A_0 \to \{A_1, \ldots, A_{N_r}\}$ *we have categorical distributions* $g_{r,i}(z|\omega)$ *defining the probability of a pose* z *for* A_i *conditional on a pose* ω *for* A_0.
6. *Self-rooting probabilities,* ϵ_A*, for each symbol* $A \in \Sigma$.

7. *A noisy-or parameter, ρ.*

The bricks *defined by $\mathcal{G}$ are pairs of symbols and poses, $\mathcal{B} = \{(A, \omega) \mid A \in \Sigma, \omega \in \Omega_A\}$.*

Definition 3.2. *A* scene *S is defined by*

1. *A set $\mathcal{O} \subseteq \mathcal{B}$ of bricks that are present in S.*
2. *A rule $r \in \mathcal{R}_A$ for each brick $(A, \omega) \in \mathcal{O}$, and a pose $z \in \Omega_{A_i}$ for each A_i in the RHS of r.*

Let $H = (\mathcal{B}, E)$ be a directed graph capturing which bricks can generate other bricks in one production. For each rule r, if $g_{r,i}(z|\omega) > 0$, we include $((A_0, \omega), (A_i, z))$ in E. We say a grammar $\mathcal{G}$ is *acyclic* if H is acyclic. A *topological ordering* of $\mathcal{B}$ is an ordering of the bricks such that (A, ω) appears before (B, z) whenever (A, ω) can generate (B, z). When $\mathcal{G}$ is acyclic we can compute a topological ordering of $\mathcal{B}$ by topological sorting the vertices of H.

Definition 3.3. *An acyclic grammar defines a distribution over scenes, $P(S)$, through the following generative process.*

1. *Initially $\mathcal{O} = \emptyset$.*
2. *For each brick $(A, \omega) \in \mathcal{B}$ we add (A, ω) to $\mathcal{O}$ independently with probability ϵ_A.*
3. *We consider the bricks in $\mathcal{B}$ in a topological ordering. When considering (A, ω), if $(A, \omega) \in \mathcal{O}$ we* expand *it.*
4. *To expand (A, ω) we select a rule $r \in \mathcal{R}_A$ according to $P(r)$ and for each A_i in the RHS of r we select a pose z according to $g_{r,i}(z|\omega)$. We add (A_i, z) to $\mathcal{O}$ with probability ρ.*

Note that because of the topological ordering of the bricks, no brick is included in $\mathcal{O}$ after it has been considered for expansion. In particular each brick in $\mathcal{O}$ is expanded exactly once. This leads to derivation trees rooted at each brick in the scene. The expansion of two different bricks can generate the same brick, and this leads to a "collision" of derivations. When two derivations collide they share a sub-derivation rooted at the point of collision. Derivations terminate using rules of the form $A \to \emptyset$, or through early termination of a branch with probability ρ.

The grammar defines a graphical model (Bayes Net), which, in turn, provides a factor graph representation. Finally, then, we can exploit the factor graph by using loopy belief propagation (LBP) for parameter estimation and inference.

To demonstrate the generality of the approach we conducted experiments with two different applications: curve detection, and face localization. Previous approaches for these problems typically use fairly distinct methods. Here, we demonstrate we can handle both problems within the same framework. In particular we have used a single implementation of a general computational engine for both applications. The computational engine can perform inference and learning using arbitrary scene grammars. We will report the speed of inference as performed on a laptop with an Intel® i7 2.5GHz CPU and 16 GB of RAM. Our framework is implemented in Matlab/C using a single thread.

Experiments in curve detection. To model binary contour maps we use a first-order Markov process that generates curves of different orientations and varying lengths. The grammar is defined by two symbols: C (oriented curve element) and J (curve pixel). We consider curves in one of 8 possible orientations. For an image of size $[n, m]$, the pose space for C is an $(n \times m) \times 8$ grid and the pose space for J is an $n \times m$ grid.

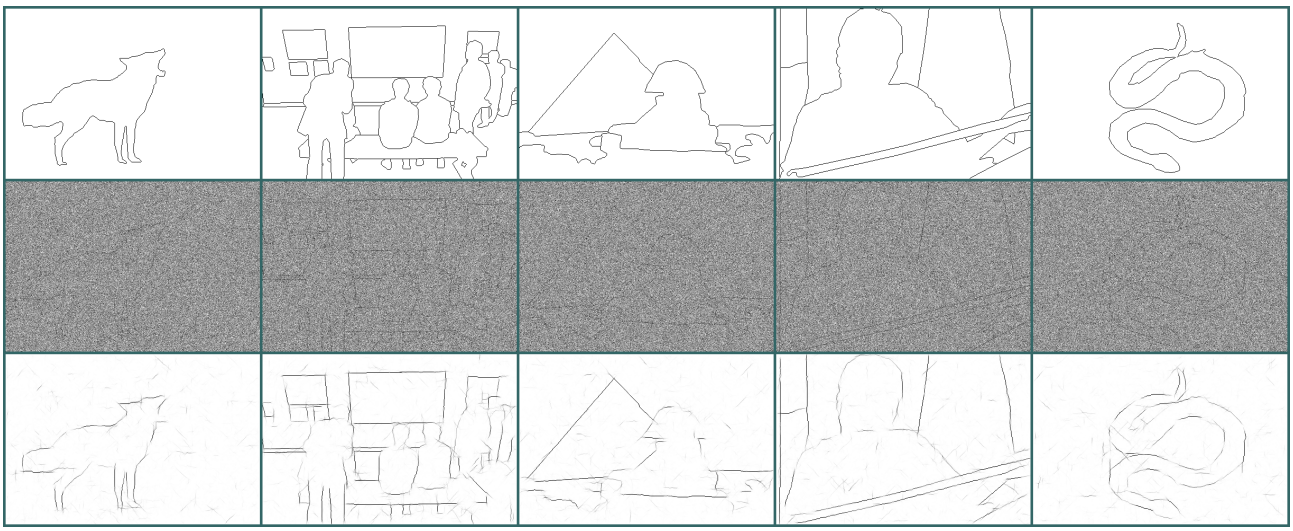

Figure 4: Curve detection results in the BSDS500 test set. Top row: Ground-truth contour maps. Middle row: Noisy observations, I. Bottom row: Estimated probability that a curve goes through each pixel, with dark values for high-probability pixels.

We can express the rules of the grammar as

$$
\begin{aligned}
&C((x,y),\theta) \rightarrow J(x,y) && 0.05\\
&C((x,y),\theta) \rightarrow J(x,y), C((x,y)+R_\theta(1,0),\theta) && 0.73\\
&C((x,y),\theta) \rightarrow J(x,y), C((x,y)+R_\theta(1,+1),\theta) && 0.11\\
&C((x,y),\theta) \rightarrow J(x,y), C((x,y)+R_\theta(1,-1),\theta) && 0.11
\end{aligned}
$$

where $R_\theta(x,y)$ denotes a rotation of (x,y) by θ. Consider generating a "horizontal" curve, with orientation $\theta = 0$, starting at pixel (x,y). The process starts at the brick $C((x,y),0)$. Expansion of this brick will generate a brick $J(x,y)$ to denote that pixel (x,y) is part of a curve in the scene. Expansion of $C((x,y),0)$ with the first rule ends the curve, while expansion with one of the other rules continues the curve in one of the three pixels to the right of (x,y).

The values on the right of the rules above indicate their (learned) probabilities. We show random contour maps J generated by this grammar in Figure 3. The model generates multiple curves in a single image due to the self-rooting parameters.

In Figure 4 we show curve detection results using the curve grammar for some examples from the BSDS500 test set. We illustrate the estimated probability that each pixel is part of a curve, $P(X(J,(x,y))=1|I)$, where I is the corrupted image. This involves running LBP in the factor graph representing the curve grammar. Inference on a (481×321) test image took 1.5 hours.

For a quantitative evaluation we compute an AUC score, corresponding to the area under the precision-recall curve obtained by thresholding $P(X(J,(x,y))=1|I)$. We also evaluate a baseline "no-context" model, where the probability that a pixel belongs to a curve is computed using only the observation at that pixel. The grammar model obtained an AUC score 0.71 while the no-context baseline achieved an AUC score of 0.11.

The use of contextual information defined by the curve grammar described here significantly improves the curve detection performance. Although our method performed well in detecting curves in extremely noisy images, the model has some trouble finding curves with high curvature. We believe this is primarily because the grammar we used does not have a notion of curvature. It is certainly possible to define more detailed models of curves.

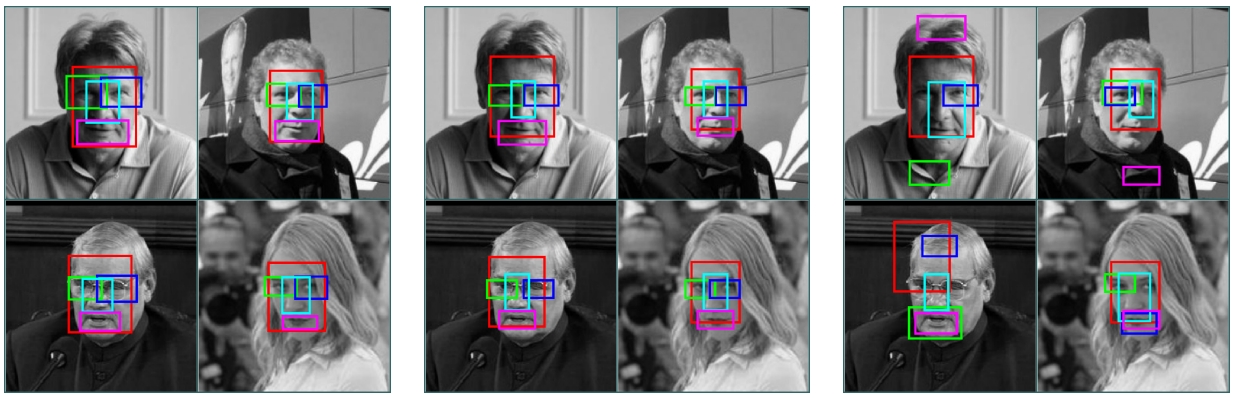

Figure 5: Localization results. Left: annotated ground-truth bounding boxes. Middle: results of the grammar model. Right: results of the baseline model using HOG filters alone. The parts are Face (red), Left Eye (green), Right eye (blue), Nose (cyan), and mouth (magenta).

Face localization. To study face localization, we performed experiments on the Faces in the Wild dataset. The dataset contains faces in unconstrained environments. Our goal for this task is to localize the face in the image, as well as face parts such as eyes, nose, and mouth. We randomly select 200 images for training, and 100 images for testing.

The face grammar has symbols Face (F), Left eye (L), Right eye (R), Nose (N), and Mouth (M). Each symbol has an associated set of poses of the form (x, y, s), which represent a position and scale in the image. We refer to the collection of $\{L, R, N, M\}$ symbols as the parts of the face. The grammar has a single rule of the form $F \rightarrow \{L, R, N, M\}$. We express the geometric relationship between a face and each of its parts by a scale-dependent offset and region of uncertainty in pose space. The offset captures the mean location of a part relative to the face, and the region of uncertainty captures variability in the relative locations. We learn the geometric parameters such as the part offsets by collecting statistics in the training data.

Figure 3 shows samples of scenes with one face generated by the grammar model we estimated from the training images in the face dataset. Note the location and scale of the objects varies significantly in different scenes, but the relative positions of the objects are fairly constrained.

Finally, the data model is based on HOG filters, which can be trained using publicly-available code. We train separate filters for each symbol in the grammar using annotated images.

Figure 5 shows some localization results. The results illustrate the context defined by the compositional rule is crucial for accurate localization of parts. The inability of the baseline (HOG-only) model to localize a part implies the local image evidence is weak. By making use of contextual information in the form of a compositional rule we can perform accurate localization despite locally weak image evidence.

We provide quantitative evaluation of the baseline ('HOG filters') and grammar ('Grammar-Full') models in the first two rows of Table 1. The Face localization accuracy of both models are comparable. However, when attempting to localize smaller objects such as eyes, context becomes important since the local image evidence is ambiguous. We also ran an experiment with the grammar model *without* a HOG filter for the face. Here, the grammar is unchanged but there is no data model associated with the Face symbol. As can be seen in the bottom row of Table 1, we can localize faces very well despite the lack of a face data model, suggesting that contextual information alone is enough for accurate face localization. Inference using the grammar model on a (250×250) test image took 2 minutes.

Model	Face	Left Eye	Right Eye	Nose	Mouth	Average
HOG filters	14.7 (18.7)	33.8 (39.7)	37.9 (35.1)	8.9 (18.1)	24.6 (35.0)	24.0
Grammar-Full	13.1 (17.1)	6.6 (12.4)	8.2 (16.5)	5.5 (10.6)	11.4 (17.7)	9.0
Grammar-Parts	13.8 (18.3)	6.1 (10.8)	8.8 (19.1)	7.4 (15.1)	12.1 (19.1)	9.7

Table 1: Mean distance of each part to the ground truth location. Standard deviations are shown in brackets. Grammar-Full denotes the grammar model of faces with filters for all symbols. Grammar-Parts denotes the grammar model with no filter for the face symbol. The grammar models significantly outperform the baseline in localization accuracy. Further, the localization of the Face symbol for Grammar-Parts is very good, suggesting that context alone is sufficient to localize the face.

4 Generative Data Models

To review, compositional models are generalizations of part-based models in which a hierarchy of part-whole relationships is formulated for the purpose of exploiting context at multiple levels of resolution. The challenge to inference lies in recognizing and propagating ambiguity until a sufficient level of context is available to make a determination, whether about a boundary, a part, an object, or a grouping of objects. Recognizing ambiguity requires a careful assessment of likelihoods. Propagating ambiguity requires maintaining an accurate approximation of the posterior distribution. A demonstration of the utility of compositional models in exploiting context can be found in the previous section.

Concerning likelihoods, the customary approach is to build probabilistic models of features rather than models of images themselves. An example is the HOG feature used to good effect in the previous section. But as suggested earlier, we have concluded that the approach will not scale. The problem with putting the distribution of features themselves arises when two features are extracted from overlapping sets of pixels; there is no way to properly normalize the joint distribution and there is then a danger (actually, an almost certainty) that we will end up comparing likelihoods defined on separate and fundamentally different spaces, precluding generative modeling. (As already mentioned in §1, these arguments do *not* apply to discriminative models, but discriminative models suffer a different and quite possibly insurmountable challenge: they will need far larger sample sizes than what is already used for *non-contextual* recognition.)

In [23] we took a first step by developing a variant of a statistical technique known as "conditional modeling" under which *pixel-level*, as opposed to *feature-level*, appearance models can be learned and sampled, and under which latent interpretations can be directly compared via likelihood ratios on the common space of pixel intensities. The approach can be used to learn mixtures of coarse or fine appearance models for image patches, and these models can be stitched together to form a single coherent data likelihood given a latent scene representation, *provided that the latent variables do not access common regions of the image*. As a concrete example, Figure 6 shows eighteen samples drawn from an eight-fold mixture model of mouths, learned from the Feret Database.

Figure 6: Samples from a compositional generative model of mouths.

Here we will briefly review the methods introduced in [23] and then present a surprisingly effective extension that maintains normalization and appears to provide strong discriminative power in a fully generative framework.

Let I represent the array of pixel values that make up an image, and let $\mathcal{G}$ be an ensemble of latent models, such as the Bayes net developed in §3. The goal is to model the distribution I through a conditional probability $p(I \mid G)$, for any $G \in \mathcal{G}$, and thereby

complete a fully (pixel-level) generative model. Under the no-overlap assumption, it is not hard to build a conditional model, $p(I \mid G)$, from local models, potentially one for every active latent structure (as represented by "bricks" in §3).

Sticking with the notation introduced in §3, let Σ be a set of types (types of bricks), such as edge, boundary, right eye, mouth, face, and so-on, and let Ω_A be a set of poses (including locations) for bricks of type $A \in \Sigma$. We will associate with every pair $(A, \omega), A \in \Sigma,\ \omega \in \Omega$, a region (collection of pixels) A_ω, and denote the corresponding vector of image intensities by I_{A_ω}. The intermediate goal is to put a conditional distribution on I_{A_ω}. We think of these local conditional distributions as "appearance models," which are necessarily more specific at the lower levels of the part-whole hierarchy (types like edges and boundaries) than at the upper levels (eyes and faces). We start with a "null" or "background" model $p^o(I)$ for the entire image. This can be an empirical distribution made up of smooth and structureless patches, learned from real images or, for the discussion here, a simple Gaussian random field (GRF) learned from these patches. Structureless patches can be easily selected, automatically, from an image library, as done in [23]. We view the null distribution as capturing many of the common properties of images that are shared across objects and background, most notably the tendency for neighboring pixels to be similar, but lacking, by construction, the detailed structure that characterizes an eye or mouth or leaf, or even something as simple as a local discontinuity or contour.

Generically, we will suppress for now the pose information and let A be an image patch, which we will take as rectangular, but may actually be of an arbitrary shape and not necessarily connected. We wish to develop a probability, $p(I_A)$, given that A is of a particular type, or category, say a "right-eye", and at a particular pose. One way to do this is through sufficiency. Given a category-specific sufficient statistic $s(I_A)$, assumed to be low dimensional, we perturb the null model to conform to the category-dependent distribution p_S on s:

$$p^o(I_A) = p^o_S(s(I_A))p^o(I_A \mid S = s(I_A)) \to p_S(s(I_A))p^o(I_A \mid S = s(I_A)) \triangleq p(I_A)$$

in which case the category-specific distribution $p(I_A)$ is the closest distribution to $p^o(I_A)$ given that the statistic $s(I_A)$ has distribution $p_S(s)$, in the sense of relative entropy, in both directions, i.e. minimizing both $D(p||p^o)$ and $D(p^o||p)$. The idea is that most of the dimensions in I_A obey generic regularity conditions of images, and that controlling a low-dimensional statistic, $s(I_A)$, is sufficient to capture the discriminating aspects of the category appearance. This, then, is an application of what is sometimes called conditional modeling in the statistics literature. A prototypical example of a sufficient statistic would be $s(I_A) = \text{corr}(I_A, T)$, where corr is the normalized correlation and T is a template to be learned from data.

More generally, we can think of $s(I_A)$ as a family of statistics, indexed by a (typically unknown) parameter ϕ, and write $s(I_A; \phi)$ in place of $s(I_A)$. The correlation statistic is then a particular example, with $\phi = T$. The distribution on s can also be parameterized, say by θ, in anticipation of learning both ϕ and θ: $p_S(s(I_A)) \to p_S(s(I_A; \phi); \theta)$.

These models can be made substantially more expressive through a generalization to category-specific mixtures, say with M components, indexed by $m = 1, 2, \ldots, M$:

$$\begin{aligned} p(I_A) &= \sum_{m=1}^{M} \epsilon_m p_{S_m}(s_m(I_A; \phi_m); \theta_m) p^o\big(I_A \mid S_m = s_m(I_A; \phi_m)\big) \\ &= p^o(I_A) \sum_{m=1}^{M} \epsilon_m \frac{p_{S_m}(s_m(I_A; \phi_m); \theta_m)}{p^o_{S_m}\big(S_m = s_m(I_A; \phi_m)\big)} = p^o(I_A) \sum_{m=1}^{M} \epsilon_m X^m_A(I_A; \phi_m, \theta_m) \quad (1) \end{aligned}$$

where X^m_A, for $m \in \{1, 2, \ldots, M\}$, is the likelihood ratio

$$X^m_A(I_A; \phi_m, \theta_m) = \frac{p_{S_m}(s_m(I_A; \phi_m); \theta_m)}{p^o_{S_m}\big(S_m = s_m(I_A; \phi_m)\big)}$$

Notice that the dependence of $p(I_A)$ on the parameters $\{\epsilon_m, \phi_m, \theta_m\}_{m=1:M}$ is only through the weighted likelihood ratios $\epsilon_m X_A^m,\ m = 1, 2, \ldots, M$, and therefore the likelihood function for the parameters depends only on these weighted ratios. Since s_m is low dimensional (one dimension in the case of normalized correlation), and since there is an unlimited supply of "background" samples (via either the empirical distribution or the GRF model), the denominator is easily evaluated as a function of the parameters. Using this observation, and a more-or-less standard modification of EM, category instances can be used to learn all of the parameters. Figure 6 shows eighteen samples drawn from an $M = 8$ component mixture model of mouths, learned using the correlation statistic from the Feret Database.

The model is flexible. By thinking of the mixture as a mixture over poses as well as over sufficient statistics, it can be used without modification to learn from unregistered data, including across scales and rotations [23]. The model can also be used to determine a subset of useful pixels in A, in essence building a mask that defines the relevant locations. Finally, we remark that whereas the distribution on the statistic (equivalently, θ) can be learned from observations of the statistic alone, the parameters of the statistic (e.g. correlation templates, or more generally ϕ) cannot be properly learned without recourse to the full pixel model.

In way of illustration, consider a detection task: distinguish eyes from background, using patches chosen randomly from the union of two collections of patches–some containing an eye and the others not containing an eye. Figure 4 compares the ROC performance of various models. All but one (namely, the Gaussian Mixture Model) use Equation (1) to compute the likelihood ratio, $\dfrac{p(I_A)}{p^o(I_A)}$, which is then thresholded to produce the (theoretically) optimal ROC under the model. In brief, "i.i.d. background" uses the trivial (and often employed) i.i.d. model for p^o; "Smooth natural background" uses an estimate of the background distribution using real images; "Gaussian random field background" fits a GRF to p^o; "Gaussian Mixture Model" is the commonly used ("default") model for image patches; and "PCA model" is a version of Equation (1) based on PCA templates.

Obviously, a standard Gaussian mixture is inadequate. As might be expected, the best of the models that use sufficiency ("conditional modeling") estimates the null distribution on the sufficient statistic from actual image backgrounds.

Consider again a scene "parse" or "interpretation," $G \in \mathcal{G}$, such as one generated by the stochastic grammars discussed in §3. To build a coherent model for the image I given the parse G, $p(I|G)$, we must "stitch together" multiple local appearance models, possibly one for every latent variable in G. The method described in the paragraphs above easily generalizes, unless some of the sufficient statistics associated with the ensemble of appearance models are functions of the same pixel intensities. As already noted, it would be a mistake to try to avoid this situation since it is natural that an annotation of a scene will include multiple annotations of certain regions.

It turns out that for the right kinds of sufficient statistics, Equation (1) has a sweeping generalization, which we now discuss.

Nonlinear filtering. The sufficient statistics we have in mind are based on an *a priori* set of locally filtered versions of I:

$$\tilde{I}_k = h_k(I), \quad k = 1, \ldots, D$$

where $\tilde{I}_k(\vec{x})$ depends only on those values of $I(\vec{y})$ for which $\vec{y}$ is close to $\vec{x}$, and where $\vec{x}$ and $\vec{y}$ are used for generic locations in the image lattice. The filter h may be linear or nonlinear, but in any case is spatially homogeneous. A prototypical example is the absolute value of the Laplacian, $\tilde{I} = |\Delta_d * I|$, where Δ_d is the discrete Laplacian. In this case, $\tilde{I}$ is useful for indicating boundary locations. Imagine that we have selected D such functions, $h_1, \ldots, h_D$, which yield D filtered images $\tilde{I}_1, \ldots, \tilde{I}_D$. Examples might include simple local averages of the intensity image, or of any of the color channels, the image I itself, absolute values of the

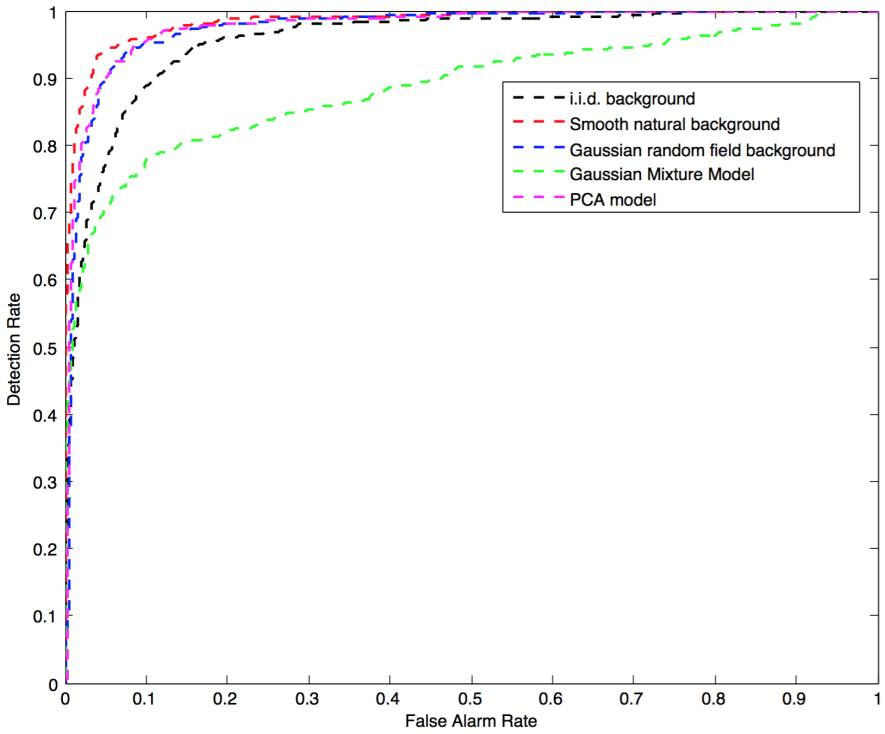

Figure 7: Eye detection. All four conditional models substantially improve on the standard Gaussian mixture model. The best model is the most realistic model. See text for details.

Laplacian, as in the example above, but with a separate function h for each of many resolutions of Δ_d, or gradient-based images, or any other filters with small spatial support.

Weberization. For every location $\vec{x}$ in the pixel array define the set $\eta_{\vec{x}}$ to be $\{\vec{y} : |\vec{y} - \vec{x}| \leq n\}$ for some positive n, let $N = |\eta_{\vec{x}}|$, and then, for every $k = 1, \ldots, D$ define

$$\mu_k(\vec{x}) = \frac{1}{N} \sum_{\vec{y} \in \eta_{\vec{x}}} \tilde{I}_k(\vec{y}), \quad \text{and}$$
$$\sigma_k^2(\vec{x}) = \frac{1}{N} \sum_{\vec{y} \in \eta_{\vec{x}}} \left(\tilde{I}_k(\vec{y}) - \mu_k(\vec{x})\right)^2$$

The collection of D images defined by

$$I_k(\vec{x}) \doteq \frac{\tilde{I}_k(\vec{x}) - \mu_k(\vec{x})}{1 + \sigma_k(\vec{x})}$$

are what we call the *Weberized* filters. They have some special properties that appear to be robust to the choice of the functions $h_1, \ldots, h_D$ and to the diameter, n, of the Weberization.

Conditional Gaussian. In particular, our experiments indicate that Weberized filters are not only marginally (nearly) stationary Gaussian random fields, but that the ensemble of Weberized filters are *jointly* (nearly) Gaus-

sian. We've tried many variants with consistent results. (The biggest departure is in the tails, which require small corrections.)

In other words, the joint distribution on the Weberized filters is fully characterized by the D means

$$m_k = E[I_k(\vec{x})]$$

where $k = 1, \ldots, D$ and $\vec{x}$ is an arbitrary pixel, and the $D(D-1)/2$ covariance functions

$$C_{k,l}(\vec{x}, \vec{y}) = E\big[\big(I_k(\vec{x}) - m_k\big)\big(I_l(\vec{x}) - m_l\big)\big] \approx C_{k,l}(|\vec{x} - \vec{y}|)$$

for all $1 \leq k \leq l \leq D$ and all pairs of pixels $\vec{x}$ and $\vec{y}$. The last, approximate, equality amounts to an isotropy assumption. It's pretty good for real images, and certainly good enough for explaining the idea. Given that the Weberized filters are (approximately) demeaned, the background (p^o) means are close enough to zero to set $m_k = 0$ for all k. The covariances, on the other hand, are not zero and need to be estimated. But isotropy and homogeneity make the task easy.

Parsing and sufficient statistics A generative latent-variable model amounts to a distribution on interpretations, $\mathcal{G}$, the ensemble of parses. What is missing is a conditional distribution on the data, $p(I|G)$, given a parse $G \in \mathcal{G}$. The conditional modeling trick suggests using sufficient statistics and examining the ratio $\dfrac{p(I|G)}{p^o(I)}$. As in Equation (1), this ratio ends up depending only on ratios of the probabilities of sufficient statistics, under the conditional distribution $p(\cdot|G)$ in the numerator and the null distribution $p(\cdot)$ in the denominator. The observations of the previous paragraphs, about families of Weberized filters, leads again to template-based sufficient statistics and their associated likelihood ratios, as follows.

Given $G \in \mathcal{G}$, let $\Sigma_G \subseteq \mathcal{G}$ be the set of types (see §3) in the parse G, one type for each participating brick. Let $n_G = |\Sigma_G|$ be the number of participating bricks and $\vec{A} = (A_1, \ldots, A_{n_G})$ be a listing of their types. Each type A_i has a pose $w_i \in \Omega_{A_i}$. Let $\vec{w} = (w_1, \ldots, w_{n_G})$, which, then, has range Ω_G, where

$$\Omega_G \doteq \prod_{i=1}^{n_G} \Omega_{A_i}$$

Associated with Ω_G is a probability distribution on the poses of the parts, $p_G(\vec{w}),\ w \in \Omega_G$. We refer, again, to §3.

As for the sufficient statistics, there is (potentially) one for every participating brick, and each is now a D dimensional vector, with one component for each of the D Weberized images:

$$\vec{S}_{A_i,w_i}(I) = \big(S_{A_i,w_i,1}(I_1), \ldots, S_{A_i,w_i,D}(I_D)\big)$$

Using the same conditioning trick as before, over and over again with different factorizations, leads to

$$\frac{p(I|G)}{p^o(I)} = \sum_{\vec{w} \in \Omega_G} p_G(w) X_G(w) \tag{2}$$

where

$$X_G(w) = \frac{p\big(\vec{S}_{A_1,w_1}(I), \ldots, \vec{S}_{A_{n_G},w_{n_G}}(I)|G\big)}{p^o\big(\vec{S}_{A_1,w_1}(I), \ldots, \vec{S}_{A_{n_G},w_{n_G}}(I)\big)} \tag{3}$$

Which is nothing more than a generalization of Equation (1). (Here we have treated the interpretation, or parse, as a mixture over the poses of the participating bricks. Depending on how the inference problem is treated,

there may also be a mixture over interpretations, via an *a posteriori* distribution on $\mathcal{G}$. This more general notion of an interpretation requires only a straightforward modification to Equation (2).)

What parametric form can we employ for the sufficient statistics that will render the numerator of (3) easy to estimate and the denominator easy to evaluate? Since the Weberized images are joint GRF's, any set of linear combinations of the pixels in the Weberized images is also jointly Gaussian. This suggests working with sufficient statistics of the form

$$S_{A_i,w_i,j} = S_{A_i,w_i,j}(I_j) =< I_j | T_{A_i,w_i,j} >, \quad 1 \leq i \leq n_G \quad 1 \leq j \leq D$$

The $T's$, then, are templates, one for each Weberized image and each pose w of each type A.

Consider, first, the evaluation of the denominator, given a set of (learned) templates $\left\{T_{A_i,w_i,j}\right\}_{1\leq i\leq n_G, 1\leq j\leq D}$. We have already noted that the mean of each Weberized image is essentially zero ($m_k^o = 0$ for all $k = 1, \ldots, D$), where the superscript indicates a parameter evaluated under p^o), and hence

$$E^o\big[S_{A_i,w_i,j}\big] = E^o\big[< I_j | T_{A_i,w_i,j} > \big] = \big[< E^o[I_j] | T_{A_i,w_i,j} > \big] = 0$$

for all $i = 1, \ldots, n_G$ and $j = 1, \ldots, D$. With the same convention about the superscript, we will denote by $C^o_{k,l}(|\vec{x} - \vec{y}|)$ the covariance of $I_k(\vec{x})$ and $I_l(\vec{y})$ under p^o, which is just $E^o\big[I_k(\vec{x})I_l(\vec{y})\big]$. In terms of these covariance functions

$$\begin{aligned} \text{COV}^o(S_{A_i,w_i,k}, S_{A_i,w_i,l}) &= E^o\big[T^t_{A_i,w_i,k} I_k I_l^t T_{A_i,w_i,l}\big] \\ &= T^t_{A_i,w_i,k} C_{k,l}(|\vec{x} - \vec{y}|) C^t_{k,l}(|\vec{x} - \vec{y}|) T_{A_i,w_i,l} \end{aligned}$$

Which completes the characterization of $p^o\big(\vec{S}_{A_1,w_1}(I), \ldots, \vec{S}_{A_{n_G},w_{n_G}}(I)\big)$, the denominator in (3). The evaluations of the data likelihoods under p^o, then, amount to convolutions, albeit a large number of them.

The model for the numerator in (3) is almost identical.

We remark that the estimation problem is made easier by noting that $T_{A_i,w_i,j}$ should differ from $T_{A_i,\tilde{w}_i,j}$ only by a change in pose, and hence only one template needs to be learned for every pair of a Weberized image and a type of brick. In fact, in terms of sample sizes, the estimation of the templates is quite efficient, requiring less than a thousand segmented human poses, extracted from street scenes, to get *what appears to be* excellent discrimination. The important disclaimer is that we can not actually run the full ROC experiment; our computation (inference) engine is not adequate, and the work on this continues. On the other hand, placing the pedestrian model, by hand, at a correct pose, followed by brute-force evaluation at nearby poses, produces a sharply peaked likelihood ratio.

In summary:

1. The data model outlined here can be expanded, arbitrarily, to add new features to any given appearance model, while maintaining normalization of the likelihood ratio.

2. Empirically and logically the evidence indicates that we now have the ability to build highly discriminative models from which it is reasonable to expect excellent ROC performance, *if we could evaluate the likelihood ratios across pose space*.

3. The problem of finding peaks in the likelihood ratio is, at this stage, a well defined algorithmic challenge and the main focus of our ongoing effort. We are exploring many directions, including modified particle filters, extensions or new efficiencies for the belief propagation methods discussed in §3, and a variety of coarse-to-fine search strategies.

5 Publications (published, submitted, or in draft form)

1. M Harrison (2012) Conservative hypothesis tests and confidence intervals using importance sampling. Biometrika, 99: 57-69.

2. S. Geman, A. Amarasingham, M. Harrison, N.G. Hatsopoulos (2012). Conditional Modeling and the Jitter Method of Spike Re-Sampling. Journal of Neurophysiology, 107(2), 517-531.

3. M. Harrison, A. Amarasingham, R. Kass (2013) Statistical identification of synchronous spiking. In, Spike Timing: Mechanisms and Function. Eds: P. Di Lorenzo, J. Victor. Taylor & Francis, 77-120.

4. L.-B. Chang, Z.-B. Bai, S.-Y. Huang, C.-R. Hwang (2013). Asymptotic Error Bounds for Kernel-based Nystrom Low-Rank Approximation Matrix. J. Multivariate Analysis, 20, 102-119.

5. M. Harrison (2013) Accelerated spike resampling for accurate multiple testing controls. Neural Computation, 25: 418-449.

6. L.-B. Chang, S. Geman (2013). Empirical Scaling Laws and the Aggregation of Nonstationary Data. Physica A, 392(20), 2013, 5046-5052.

7. J. Miller, M. Harrison (2013). Exact sampling and counting for fixed-margin matrices. Annals of Statistics. Annals of Statistics. Vol. 41, No. 3, pp. 1569-1592.

8. T.-L. Chen, S. Geman. Image Warping Using Radial Basis Functions (2013). Journal of Applied Statistics. J. Applied Statistics, 41(2), 242-258.

9. L.-B. Chang, S. Geman, F. Hsieh, C.-R. Hwang (2013). Invariance in the Recurrence of Large Returns and the Validation of Models of Price Dynamics. Physical Review E, 88(2), 2013, 022116-1:022116-15.

10. . J. Miller, M. Harrison (2013). A simple example of Dirichlet process mixture inconsistency for the number of components. Advances in Neural Information Processing Systems NIPS), 26.

11. M. Homer, M. Harrison, M. Black, J. Perge, S. Cash, G. Friehs, L. Hochberg (2013) Mixed decoded cursor velocity and position from an offline Kalman filter improves cursor control in people with tetraplegia. Proceedings of the 6th International IEEE EMBS Conference on Neural Engineering.

12. Bevan Keeley Jones, Sharon Goldwater, and Mark Johnson (2013). Modeling Graph Languages with Grammars Extracted via Tree Decompositions. In Proceedings of the 11th International Conference on Finite State Methods and Natural Language Processing, pages 54-62.

13. M. Homer, J. Perge, M. Black, M. Harrison, S. Cash, L. Hochberg (2014). Adaptive offset correction for intracortical brain computer interfaces. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 22: 239-248.

14. D. Geman, S. Geman, N. Hallonquist, L. Younes (2015). Visual Turing test for computer vision systems. PNAS, vol. 112(12), 3618-3623.

15. A. Amarasingham, M. Harrison and S. Geman (2015). Ambiguity and non-identifiability in the statistical analysis of neural codes. PNAS, PNAS, vol. 112(20), 2015, 64556460.

16. L.-B. Chang, D. Geman (2015). Tracking cross-validated estimates of prediction error as data accumulate. JASA, (in press)

17. W. Truccolo, O. Ahmed, M. Harrison, E. Eskandar, R. Cosgrove, J. Madsen, N.S. Potter, L. Hochberg, S. Cash (2014). Neuronal ensemble synchrony during human focal seizures. Journal of Neuroscience.

18. J. Miller, M. Harrison (2014). Inconsistency of Pitman-Yor process mixtures for the number of components. Journal of Machine Learning Research, vol 15(1), 3333-3370.

19. M. Harrison, A. Amarasingham, W. Truccolo (2015) Spatio-temporal conditional inference and hypothesis tests for neural ensemble spiking precision. Neural Computation, vol 27(1), 104-150.

20. M. Harrison (2014). Significance evaluation. In, Encyclopedia of Computational Neuroscience. Eds: D. Jaeger, R. Jung. Springer.

21. G. Zhou, S. Geman, and J. Buhmann (2014). Sparse feature selection by information theory. Proceedings of the 2014 IEEE International Symposium on Information Theory, 926-930.

22. Matthew T. Harrison, Asohan Amarasingham, and Wilson Truccolo (2015). Spatiotemporal conditional inference and hypothesis tests for neural ensemble spiking precision. Neural Computation 27:1, 104-150.

23. L.-B. Chang, E. Borenstein, W. Zhang, S. Geman (2015). Maximum likelihood features for generative image models. (accepted, pending revision)

24. J. Chua (2014). Inference in Grammer-based Models of visual context Using Approximations of Approximations. (draft)

25. J. Miller (2014). Nonparametric Bayes. Ph.D. Thesis, Division of Applied Mathematics, Brown University.

26. N. Hallonquist (2016). Random Graph Modeling and Discovery. Ph.D. Thesis, Department of Applied Mathematics, Johns Hopkins University.

27. D. Geman, N. Hallonquist, and L. Younes (2016). Scene modeling for visual Turing tests. Working paper, Center for Imaging Sciences, Johns Hopkins University.

28. J. Chua and P. Felzenzwalb (2016). Scene Grammars, Factor Graphs, and Belief Propagation. Submitted for publication.